Prototyping a Novel Augmented-Reality Head-Mounted Display

Sam Girvin
Stanford University
sgirvin@stanford.edu

Olivier Jin
Stanford University
ojin@stanford.edu

Anna Zeng
Stanford University
annazeng@stanford.edu

Abstract

Pass-through augmented reality has gone largely unnoticed in today's fervor for augmented reality. To discover if this approach to AR is suitable for a general population, we built a hardware and software system that demonstrates a novel, real-time pass-through AR system. The untethered system is comprised of a small OLED screen, an RGB-D camera, and a Raspberry Pi in a modified Google Cardboard housing.

1. Introduction

Augmented reality (AR) can be defined as ”a real-time direct or indirect view of a physical real-world environment that has been enhanced / *augmented* by adding virtual computer-generated information” [1]. According to Milgram's seminal survey of mixed reality devices, augmented reality is an experience that is closer to the real world than traditional virtual reality [6], marked by enhancing reality as opposed to allowing reality to enter a virtual environment.

Met with an incredible surge in academic, commercial, and government interest, AR has become one of the fastest-growing areas in modern technology. Applications in education [3], entertainment and advertising, medicine [10], and personal computing [1] have opened up new excitement in this field. Compared to traditional virtual reality (VR) displays, AR carries the added benefit of incorporating virtual elements into a real-world setting. As with VR, AR creates an immersive experience that opens up a new dimension of human interaction [3].

To create a realistic augmented reality experience, many vendors [1] have adopted the model of using a high-power commercial PC to computationally power and render an AR experience. While this allows the AR experience to provide more complex, visually accurate virtual object representations, overall this creates a host of challenges that slow AR's adoption [3].

One notable barrier to successful adoption and integration of AR is input latency. Too high of an input latency results in motion sickness and an ultimate loss of immersion for the user. Some sources of such latency include:

- **Off-Host Delay**: delay between a physical event and occurrence to a host machine
- **Computational Delay**: time spent in data processing
- **Rendering Delay**: time spent in image generation
- **Display Delay**: delay between sending an image to a display and actually displaying it
- **Synchronization Delay**: time spent to synchronize state between a host machine and AR headset
- **Frame-rate Induced Delay**: a low frame-rate will allow the viewer to constantly see an outdated image

Each of these sources of delay have been individually addressed with graphics and systems optimizations [2]; time-critical computing, parallelization, movement prediction, and post-rendering warping have been able to reduce the overhead for supporting an immersive AR experience.

One surprisingly effective, simple way to reduce this latency is to use an approach called pass-through AR. Pass-through AR adds digital elements into the users field of view, to create the visual illusion of incorporating these elements into the physical world. This technique can be done by covering one of the users eyes with a monocular display. The users brain will still be able to perform binocular fusion using the camera and the users uncovered eye. However, the viewer can also experience additional information from the screen overlaid on the real world in front of them. This overlay effect is referred to as binocular rivalry.

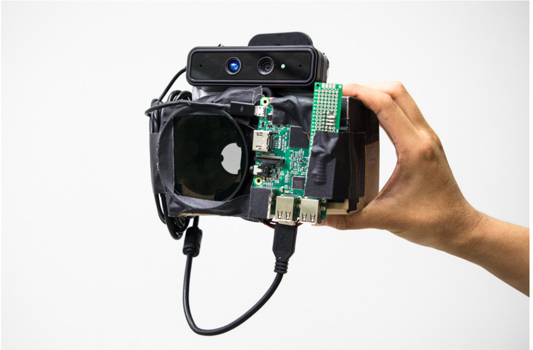

Figure 1: Completed AR Headset

2. Background and related work

The use of monocular displays is not a new concept. As far back as 2003, displays such as Microvisions Nomad product line were already relying on monocular displays to relay virtual information to the user. [2] However, the issues with monocular displays have been well-documented in previous papers. For example, Eli Peli noted that the displayed images appeared to move even without user head motion, due to small eye movements caused by the vestibulo-ocular reflex [8].

Previous work in optical rivalry [4] and binocular rivalry [9] seem to indicate that monocular displays prove to be a challenge for certain segments of the population, especially those with aging eyes. A notable example of pass-through AR that may be able to combat optical rivalry is the Eyetap [5]. In this monocular display, the display uses a small camera near the display to render what the eye would naturally see (in HDR) and reduce the mental load of binocular fusion.

With our monocular display populated with colored rectangles on a completely black screen, our headset sets the stage for a thorough exploration in the effectiveness of this pass-through AR approach.

3. Building the AR HMD

3.1. Hardware Components

The headset is comprised of three main components: a SoftKinetic DepthSense DS325 camera to collect information from the real world, a Raspberry Pi 3 to process the information, and an Adafruit 128x96 pixel RGB OLED to display final outputs. The DS325 is connected to the Raspberry Pi via USB, and the Pi connects to the OLED display over a SPI bus. Since everything can be powered by a USB battery pack, the headset can be operated completely untethered.

The frame of the headset is a modified Google Cardboard. The left half of the Cardboard is light-tight and contains the OLED screen. The OLED is positioned such that it lies in the center of the left eye's field of view, on the same plane that a phone's screen would occupy inside of the Cardboard. Light-tight black gaffer's tape carefully lines the left eye cavity, so no light can filter in; the left eye's viewing cavity must be as dark as possible, to reduce any distracting light artifacts to distract the viewer. The right eye's viewing cavity, by contrast, has a large hole cut out of it to allow the user to see the outside world. To reduce the sharp difference in brightness between left and right eyes, we added a filter to cut down on the incident light. Any neutral density filter would have worked well; in our case, we used a standard 74mm circular polarizing filter as it was the filter we had on hand that performed the best. The DS325 and battery sit on top of the housing. The Raspberry Pi is attached over the left eye, on top of the OLED screen where it is out of the way.

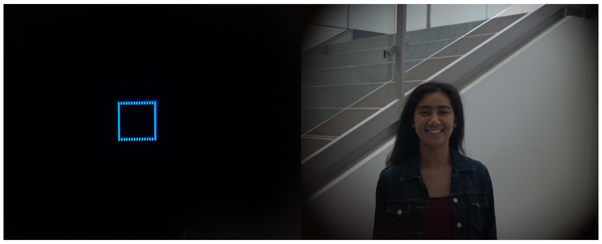

Figure 2: The view from the two different eyes, as captured by a digital camera.

The packaging of the headset was optimized for portability and discussion; all wires are tucked away on the side opposite to the Raspberry Pi to better place the center of gravity. The heaviest components of the device – the DepthSense camera and the USB battery pack – were placed on the top of the Cardboard to help laterally adjust the center of gravity and prevent accidental disconnections. The slightly top-heavy device would translate to a firm, comfortable feeling of presence to the AR headset wearer when the neck of the DepthSense camera rests on the wearer's forehead. As the OLED screen is behind a small Cardboard lens and all other electronics are placed outside of the Cardboard housing, the wearer is protected even in the case of an electrical malfunction.

3.2. Software Configuration

Similar to the hardware, the software components can be broken down into four corresponding interfaces: the DepthSense libraries, the Raspberry Pi libraries, the Arduino libraries, and the OpenCV libraries.

The **DepthSense Interface** retrieves information from the camera and sends it to OpenCV for processing. The DS325 itself has multiple different operating modes, and can capture RGB, depth, and audio channel inputs. Each of these receivers can be activated separately by registering the corresponding camera node and streaming data directly into an application via a colorMap/depthMap object.

In order to do so, the DepthSense software developer kit (SDK) first had to be integrated with the Linux ARM distribution installed on the Raspberry Pi (running on Raspbian). Setting up this step required emailing SoftKinetic MVP Mitch Riefel and acquiring a previously-unreleased distribution of the SDK. Once fully set up, the DepthSense camera could stream information into the Raspberry Pi via one of the Pi's USB ports.

While the Depthsense DS325 captures camera inputs, the **OpenCV libraries** perform the actual processing for facial recognition. OpenCV includes native support for face detection via their `face` libraries located in `opencv_contrib`. By instantiating a `FaceRecognizer` object and assigning a Haar cascade classifier [7], the `FaceRecognizer` could be trained to classify a variety of faces. In this case, the `FaceRecognizer` used a `frontal_face` classifier to recognize faces turned toward the camera. In each call to the classifier, we would feed in a grayscale frame of video from the DepthSense camera in order to find portions of the image marked as faces.

The Adafruit OLED for this project is driven by a Solomon Systech SSD1351. There were no existing Raspberry Pi libraries to interface with the SSD1351, so we ported the Adafruit Arduino libraries for graphics and for driving the SSD1351. This involved removing all Arduino-specific inheritance and including the use of WiringPi, a Raspbian library that allows access to the Raspberry Pi's GPIO header pins and SPI/I2C busses, to work with the OLED. The ported libraries streamline rendering simple shapes to the OLED quickly; in our code, we made use of optimized functions to draw and erase lines to reduce latency.

Furthermore, after connecting these three separate components into one working program, we calibrated the camera and OpenCV's output to align with reality. The mismatch between DepthSense camera's 720p resolution and the OLED screen's 128x96 pixel resolution was enough of a difference to compel some fine-grained adjustment, with some implicit assumptions about interpupillary distance (IPD), to create our final effect.

One of the difficulties in configuring the software environment was ensuring that all dependencies were met. Since the final program needed to interface with all four libraries, the only common programming language was C++. `Cmake` helped us include complex OpenCV dependencies with our other library dependencies by dynamically creating the Makefile based on specified input libraries and directories.

4. Evaluation/Results

Once completed, the headset performed as specified. The DepthSense camera was able to detect faces up to a range of 15 feet, and the headset could correctly categorize faces as being close (red), intermediate (yellow), or far (cyan). While in use, the viewer would notice that faces were framed by the rectangles drawn on the OLED display. The system had a refresh rate of 2-3Hz, creating a noticeable delay between head movement and frame updates.

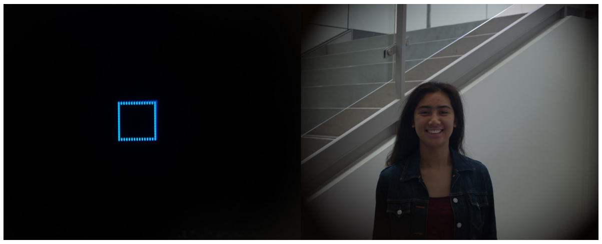

Figure 3: The view from both eyes side-by-side, both captured by a digital camera

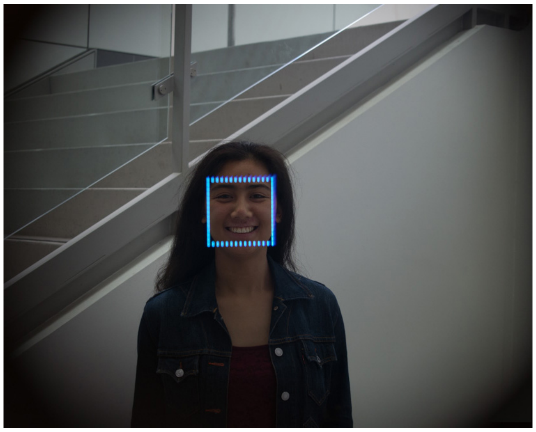

Figure 4: A rendering of the image intended to be seen

However, not all viewers were able to successfully experience binocular fusion. A few participants noted that the drawn rectangles were positioned at an offset from the actual face, with estimated offsets ranging 1-15 centimeters.

This offset could be caused by differences in volunteer IPD. Viewers with significantly different IPDs may experience difficulties in binocular fusion, since the OLED display would not align with real-world images without additional calibration. As a result, further user studies and more integration of camera data are needed to address these root causes and design a solution for the next iteration of hardware.

Another issue that arose was that several volunteer members reported difficulties in focusing on both the rectangle and the face simultaneously. However, after some time, these viewers became more accustomed to this binocular fusion, suggesting an adjustment period must elapse before the user adapts to using the headset.

5. Future Work

5.1. Hardware and Software Improvements

While the headset is able to recognize faces successfully, many improvements can still be made to the current system.

One of the limitations of the current headset is its rigidly-defined frame, which worked well for certain viewers but caused difficulties with others as described above. This issue can be resolved by placing the OLED display on an adjustable track or set of rails, so it can be shifted from left to right by a knob the viewer controls. With this change, the OLED can be configured based on the IPD of the viewer. Some viewers noted that slightly shifting the headset left or right improved their experience using the headset. As a result, an adjustable track would let each viewer manipulate the OLED to provide an optimum viewing experience for themselves.

Another mechanical change would be to add a second polarizer on top of the first, mounted on top of a set of freely-rotating rails. This would viewers to adjust the level of light intake in the right-side viewport. As described above, dimming the light received by the right eye creates a more immersive environment for the viewer. Therefore, by adding a second polarizer, the viewer could be able to vary the amount of light entering the viewport simply by twisting the second polarizer.

Currently, the face recognition system is sensitive and abrupt. Adding noise filtering on the position and movement of tracked faces could go a long way towards making the face tracking feel more organic and comfortable for users. The classifier is prone to misidentifying faces when presented with emergency signs, posters, and door frames in dimmer environments, so adding some minor image processing before feeding the video frame into the classifier can improve the experience.

Finally, the DS325 camera could be upgraded (or at least configured) to achieve better performance. One issue with the current camera is its tendency to initialize images upside-down, resulting in incorrect coordinates when identifying faces. This problem is compounded by the fact that, when fully wireless, the viewer can only reset the program by reconnecting to a workstation and restarting the face recognition software. Using a different camera might resolve this initialization issue and produce images that are always right-side up, saving end-users a significant hassle.

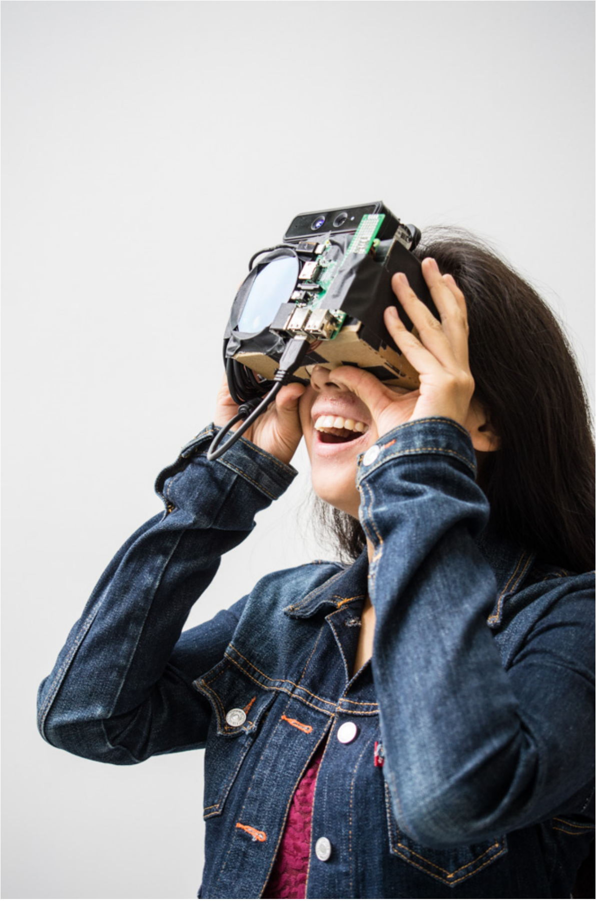

Figure 5: There are many possibilities for this sort of AR display.

5.2. Potential User Studies

In addition to potential future hardware improvements, there are also a number of user studies that could be done with monocular AR displays of this type.

Not everybody who used the headset was able to fuse the left and right eye views. This could potentially be due to a number of reasons. Some candidates are left/right eye dominant, have presbyopia or myopia, use corrective lenses, and/or have weaker control over vergence of the eyes. We can explore the effect these eye characteristics have in binocular fusion.

Another interesting experiment would be to explore viewer's perception of text on the OLED display. Text display is extremely important for any sort of informational display; however, binocular display of text in AR is difficult because it is hard to align such high density information well for each eye. Monocular display of text could remove the alignment problem, but could possibly introduce new issues.

The quantity of information displayed on the OLED was deliberately kept low (e.g. small rectangles) during our initial testing in order to make sure that binocular fusion could replace the darkness of the left eye with information from the right eye. Understanding the limits of how much information can be displayed in the dark eye view would be very important for future implementations of this sort of AR display.

How bright does the information display on the OLED need to be in order to fuse properly? Does the necessary brightness change under different ambient lighting conditions? Does the effectiveness of the display change under bright sunlight or darkness?

6. Conclusion

This paper describes an approach for pass-through augmented reality using a depth camera and OLED display and begins to explore its practicality for a general audience. One application, facial recognition in openCV, is shown and demonstrated by using rectangles of various shapes and colors to delimit facial boundaries. These rectangles are then projected onto an OLED screen, allowing binocular fusion by the viewer to superimpose the boundaries around the faces of human subjects. The headset was successfully tested by multiple volunteers at Stanford EE267's Demo day, and establishes a solid foundation for further work into affordable, efficient, and effective augmented reality technologies.

7. Acknowledgements

Many thanks to Professors Gordon Wetzstein (Stanford) and Steve Mann (Univ. Toronto) for providing guidance for this project. We would also like to thank the Wetzstein lab for lending us hardware to make this possible and Stanford's lab64 for providing space for us to work. Additional thanks to Mitch Reifel of SoftKinetic for providing us with the ARM development SDK for the DS325 and responding to our inquiries with lightning speed.

References

[1] J. ”Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic. ”augmented reality technologies, systems and applications”. *”Multimedia Tools and Applications”*, ”51”(”1”):”341–377”, ”2011”.

[2] M. C. Jacobs, M. A. Livingston, and A. State. Managing latency in complex augmented reality systems. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D ’97, pages 49–ff., New York, NY, USA, 1997. ACM.

[3] L. Kerawalla, R. Luckin, S. Seljeflot, and A. Woolard. ”making it real”: Exploring the potential of augmented reality for teaching primary school science. *Virtual Reality*, 10(3-4):163–174, 2006.

[4] A. E. Kertesz and H. J. Lee. Comparison of Simultaneously Obtained Objective and Subjective Measurements of Fixation Disparity. *American Journal of Optometry and Physiological Optics*, 64(10):734–738, 1987.

[5] S. Mann, J. Fung, C. Aimone, A. Sehgal, and D. Chen. Designing eyetap digital eyeglasses for continuous lifelong capture and sharing of personal experiences. *Alt. Chi, Proc. CHI 2005*, 2005.

[6] P. Milgram and F. Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994.

[7] OpenCV Dev Team. Cascade Classifier, 2017.

[8] E. Peli. Visual issues in the use of a head-mounted monocular display. *Optical Engineering*, 29(8):883–892, 1990.

[9] K. Ukai, H. Ando, and J. Kuze. Binocular rivalry alternation rate declines with age. *Perceptual and Motor Skills*, 97(2):393–397, 2003.

[10] H.-K. Wu, S. W.-Y. Lee, H.-Y. Chang, and J.-C. Liang. Current status, opportunities and challenges of augmented reality in education. *Computers and Education*, 62:41 – 49, 2013.